

PATENT ABSTRACTS OF JAPAN

Ref. ①

(11)Publication number : 2003-030609

(43)Date of publication of application : 31.01.2003

(51)Int.Cl.

G06K 19/07
G06F 9/445

(21)Application number : 2001-212023

(71)Applicant : DAINIPPON PRINTING CO LTD

(22)Date of filing : 12.07.2001

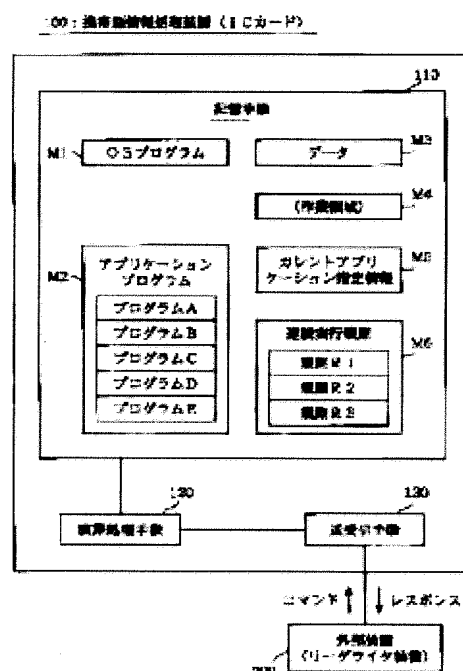
(72)Inventor : JINRIKI TETSUO
IRISAWA KAZUYOSHI
SHIBATA NAOTO

(54) PORTABLE INFORMATION PROCESSOR

(57)Abstract:

PROBLEM TO BE SOLVED: To easily add a new function for a specified application stored in an IC card.

SOLUTION: A plurality of applications A-E are prepared in an area M2 inside the IC card 100, a current application is specified by supplying an application selection command from the side of an external device 200 and specifying information is recorded in an area M5. A normal execution command is delivered from an OS program to the current application and executed. As a special command, a successive execution command to be executed over a plurality of the applications is prepared and a successive execution rule is stored inside an area M6 for each successive execution command. In the case that the successive execution command is supplied, a routine over a plurality of the applications is executed in a prescribed order on the basis of the successive execution rule.



[Claim(s)]

[Claim 1]it carrying as a simple substance at the time of carrying, being carried out, and in a portable information processor constituted so that a predetermined response might be returned to a predetermined command given from an external device at the time of access, The 1st field where OS program was recorded, and the 2nd field where two or more application programs were recorded, The 3rd field that stores data, and the 4th field used as workspace, The 5th field that stores current application specification information that current application selected from said two or more application programs is specified, The 6th field that stores a continuous execution rule which shows what two or more routines which straddled said two or more application programs should be performed for in the given order, While receiving a memory measure which ****, an arithmetic processing means which executes said application program under management of said OS program, and a command given from said external device and handing over to said arithmetic processing means, a time of an application selection command to have a transmission and reception means which performs processing which transmits a response given from said arithmetic processing means to said external device, and for said arithmetic processing means specify current application being given -- or When specific current application is specified by directions of OS program, When processing which stores in said 5th field current application specification information which shows this current application is performed and an execute command for single applications is given, Using current application specification information stored in said 5th field. When processing which executes a given command is performed based on an application program specified at the time as current application and a continuous execution command for two or more applications is given, A portable information processor having the function to perform processing which performs two or more routines in the given order, based on a continuous execution rule stored in said 6th field.

[Claim 2]In the portable information processor according to claim 1, a continuous execution rule adds to a rule which shows an order of a purport that the 2nd routine should be performed after execution of the 1st routine, A portable information processor including a rule of a purport that a parameter generated by execution of the 1st routine is handed over to the 2nd routine.

[Claim 3]In the portable information processor according to claim 2, two or more candidates who should become the 2nd routine are provided by a continuous execution rule, A portable information processor including a rule of a purport that one routine in said two or more candidates should be chosen as the 2nd routine, and this should be performed, based on a parameter generated by execution of the 1st routine.

[Claim 4]In the portable information processor according to any one of claims 1 to 3,

when an execute command for single applications or a continuous execution command for two or more applications is given from a transmission and reception means, an arithmetic processing means, Based on an application program specified at the time as current application, without distinguishing both, A portable information processor recognizing a given command to be a continuous execution command for two or more applications, and performing processing based on a continuous execution rule when processing which tries execution of a given command is performed and an error arises.

[Claim 5]In the portable information processor according to any one of claims 1 to 4, two or more continuous execution rules are stored in the 6th field, and an arithmetic processing means, A portable information processor performing processing which performs two or more routines in the given order based on a continuous execution rule corresponding to a given continuous execution command for two or more applications.

[Claim 6]A portable information processor performing processing which stores said new continuous execution rule in the 6th field when a continuous execution rule addition command in which an arithmetic processing means includes a new continuous execution rule is given in the portable information processor according to claim 5.

[Detailed Description of the Invention]

[0001]

[Field of the Invention]about a portable information processor like an IC card, especially, this invention is carried as a simple substance at the time of carrying, is carried out, and relates to the portable information processor constituted so that a predetermined response might be returned to the predetermined command given from the external device at the time of access.

[0002]

[Description of the Prior Art]The role which the portable information processor in community at large bears is also becoming still more important as the miniaturization of electronic equipment progresses. In various fields, especially the IC card that should also be called representation of a small portable information processor is put in practical use in recent years, and is expected to continue to spread widely. A portable information processor like this IC card is usually carried in connection with the body, vehicles, etc., and access from the outside is attained by making it communicate with the reader writer device connected to the personal computer etc. if needed.

[0003]The IC card which has generally spread now is a thing of the type which contained CPU.

Access to an internal memory is possible via built-in CPU.

For this reason, practically sufficient security is secured. Generally, the exchange

between an IC card and an external device is performed by repeating the process of transmitting a command from the external device side to an IC card, and transmitting a response from an IC card to an external device as a reply to this command. In an IC card, OS program for usually carrying out generalization control of the fundamental function as an information processor and the application program executed under management of this OS program are prepared. OS program and an application program perform a certain processing based on the command given from the external device, and are common in [that it is a program] that processing which returns a certain response is performed.

[0004]These days, the storage capacity of the memory built in an IC card is also increasing.

The IC card in which two or more application programs other than OS program were stored is also used increasingly widely.

It enables it to store such an application program in the nonvolatile memory of EEPROM etc. which can be written in, and a usage pattern which carries out additional installation of the required application program later is also spreading. Thus, when giving a command and performing a certain processing to the IC card for which two or more application programs were prepared, processing which gives an application selection command to the IC card concerned, and makes specific application a preparatory state beforehand is performed. Generally, the application which was in the preparatory state by the application selection command is called "current application." By OS program, at the time, the execute command for applications given from the external device side is handed over by the application program which is a preparatory state as "current application", and is executed.

[0005]

[Problem(s) to be Solved by the Invention]As mentioned above, the IC card of one sheet can be used for various uses by installing two or more application programs in the IC card. Since it is also possible to carry out additional installation of the new application program if needed, a new use can also be added. However, in the conventional IC card, it is difficult to add an additional function to the already installed application program. This is because it becomes difficult to add the routine for a new function since the program area for the application concerned, a data area, etc. will be become final and conclusive on a memory if one application program is installed. Therefore, when an option needs to be added to one application, an exchange colander is not obtained to a new program, but revised large-scale work is [whole application program] needed in the conventional IC card, for it.

[0006]Then, in the portable information processor which stored two or more application

programs, an object of this invention is to provide the new composition which can add a new function easily about specific application.

[0007]

[Means for Solving the Problem](1) carrying the 1st mode of this invention as a simple substance at the time of carrying, it being carried out, and in a portable information processor constituted so that a predetermined response might be returned to a predetermined command given from an external device at the time of access, The 1st field where OS program was recorded, and the 2nd field where two or more application programs were recorded, The 3rd field that stores data, and the 4th field used as workspace, The 5th field that stores current application specification information that current application selected from two or more application programs is specified, The 6th field that stores a continuous execution rule which shows what two or more routines which straddled two or more application programs should be performed for in the given order, While receiving a memory measure which ****, an arithmetic processing means which executes an application program under management of OS program, and a command given from an external device and handing over to an arithmetic processing means, a transmission and reception means which performs processing which transmits a response given from an arithmetic processing means to an external device being established, and an arithmetic processing means, when an application selection command for specifying current application is given, When specific current application is specified by directions of OS program, When processing which stores in the 5th field current application specification information which shows this current application is performed and an execute command for single applications is given, Using current application specification information stored in the 5th field. When processing which executes a given command is performed based on an application program specified at the time as current application and a continuous execution command for two or more applications is given, It is made to have the function to perform processing which performs two or more routines in the given order, based on a continuous execution rule stored in the 6th field, and enables it to realize an additional function by using a routine of another application program.

[0008](2) In a portable information processor which the 2nd mode of this invention requires for the 1st above-mentioned mode, As a continuous execution rule, a rule of a purport that a parameter which is generated by execution of the 1st routine after execution of the 1st routine in addition to a rule which shows an order of a purport that the 2nd routine should be performed is handed over to the 2nd routine is used.

[0009](3) In a portable information processor which the 3rd mode of this invention requires for the 2nd above-mentioned mode, As a continuous execution rule, two or more

candidates who should become the 2nd routine are set, and a rule of a purport that one routine in two or more candidates should be chosen as the 2nd routine, and this should be performed is used based on a parameter generated by execution of the 1st routine.

[0010](4) In a portable information processor which the 4th mode of this invention requires for the above-mentioned 1st - the 3rd mode, When an execute command for single applications or a continuous execution command for two or more applications is given from a transmission and reception means, an arithmetic processing means, Based on an application program specified at the time as current application, without distinguishing both, When processing which tries execution of a given command is performed and an error arises, a given command is recognized to be a continuous execution command for two or more applications, and it is made to perform processing based on a continuous execution rule.

[0011](5) In a portable information processor which the 5th mode of this invention requires for the above-mentioned 1st - the 4th mode, Two or more continuous execution rules are stored in the 6th field, and an arithmetic processing means enables it to ** performing processing which performs two or more routines in the given order based on a continuous execution rule corresponding to a given continuous execution command for two or more applications.

[0012](6) The 6th mode of this invention enables it to perform processing which stores a new continuous execution rule in the 6th field in a portable information processor concerning the 5th above-mentioned mode, when a continuous execution rule addition command in which an arithmetic processing means includes a new continuous execution rule is given.

[0013]

[Embodiment of the Invention]Hereafter, it explains based on the embodiment illustrating this invention. Drawing 1 is a block diagram showing the composition of the portable information processor (specifically IC card) 100 concerning one embodiment of this invention. This portable information processor 100 is carried as a simple substance at the time of carrying, and is connected to the external device (specifically reader writer device) 200 at the time of access. If a predetermined command is given from this external device 200, predetermined processing will be carried out in portable information processor 100 inside, and that result will be returned as a response to the external device 200.

[0014]The basic building blocks of the portable information processor 100 are the memory measure 110, the arithmetic processing means 120, and the transmission and reception means 130 as a graphic display. Various programs and data are recorded on the memory measure 110. The arithmetic processing means 120 executes the command given

from the external device 200 based on the program within this memory measure 110. Transmitting and receiving processing to the external device 200 is performed by the transmission and reception means 130. The contact method which contacts an electrode physically and exchanges an electrical signal may perform communication between the transmission and reception means 130 and the external device 200, and the noncontact method which forms an electromagnetic signalling channel and exchanges information may perform it. It is received by the transmission and reception means 130, and the command given from the external device 200 is executed by the arithmetic processing means 120. A certain response is outputted from the arithmetic processing means 120 after execution of a command, and it is transmitted to the external device 200 by the transmission and reception means 130. Thus, communication between the portable information processor 100 and the external device 200 is performed by repeating the command transmission from the external device 200, and the response transmission from the portable information processor 100 by turns.

[0015]In the memory measure 110, six fields are provided as the graphic display. OS program for carrying out generalization management of the processing operation by this portable information processor 100 is stored in the 1st field M1. Fundamentally, the arithmetic processing means 120 will perform various processings under management of this OS program. Two or more application programs are stored in the 2nd field M2. By a diagram, the example for which five kinds of application programs A-E are prepared is shown concretely. Usually, each of these application programs turn into a program used for a use different, respectively. For example, the program A is a program to perform the processing for using this IC card as an examination card for hospitals, and the program B. It is a program which performs processing for using this IC card as a credit card, and the program C is a program which performs processing for using this IC card as a loan card for libraries, and is the condition --.

[0016]Data is stored in the 3rd field M3. This data is data for OS program and each application program to use, there is data used in common from two or more programs, and it has data peculiar to each application program. Although it is common practically to store in the field contiguous to the storing region of the application program concerned as for data peculiar to a specific application program, Here, the data used with which application also gives the following explanation for convenience as what is stored in the 3rd field M3. The 4th field M4 is a field used as workspace when the arithmetic processing means 120 performs various data processing, and the 5th field M5 is a field for storing current application specification information. Here, "current application" is one application selected from two or more application program A-E stored in the 2nd field M2, and it will be called "the application chosen at present." The command given from

the external device 200 is executed by this "current application" in principle (as mentioned later, as long as it is an execute command for single applications). Although it is also possible to deal with it as a part of 4th field M4, since this 5th field M5 is an important component in this invention, it is made to define the 5th field M5 as a field for exclusive use for storing "current application specification information" here.

[0017]The 1st field M1 - 5th field M5 that were mentioned above are a field used also in the conventional IC card. Further, the feature of this invention forms the 6th field M6, stores the continuous execution rules R1-R3 here, and is at the point of having enabled it to perform two or more routines which straddled two or more application programs currently prepared in the 2nd field M2. The continuous execution rules R1-R3 are rules which show what two or more routines which straddled two or more application programs, respectively should be performed for in the given order so that it may explain in full detail behind here, The arithmetic processing means 120 can make two or more applications process the command given from the outside based on this continuous execution rule.

[0018]Although one block shows the memory measure 110 in the embodiment shown in drawing 1, Actually, in the IC card, nonvolatile memory, such as ROM and EEPROM, and volatile memory, such as RAM, are built in, it distributes in the memory of these plurality suitably, and each fields M1-M6 are established in it. According to this embodiment, the 1st field M1 that stores OS program is formed in ROM (a part is provided in EEPROM and it may be made to be possible [rewriting of it]). The 5th field M5 that stores the 4th field M4 and current application specification information that are used as workspace is formed in RAM. In this IC card, since all electric power is supplied from the external device 200 side, when communication with the external device 200 is completed and the external device 200 is separated, all the contents of the 4th field M4 and the 5th field M5 will be lost. On the other hand, the 2nd field M2 that stores an application program, the 3rd field M3 that stores data, and the 6th field M6 that stores a continuous execution rule are formed by each in EEPROM. Therefore, these information will be held as they are, even after the external device 200 is separated. Since rewriting and additional writing are possible for EEPROM, each of application programs, data, and continuous execution rules can be corrected or added.

[0019]The arithmetic processing means 120 comprises a CPU for IC cards, and has the function to perform application program A-E under management of OS program. As mentioned above, the command given from the external device 200 is handed over by the arithmetic processing means 120 via the transmission and reception means 130. Generally, a command can be classified into the command (henceforth the execute command for OS program) for performing OS program, and the command (henceforth

the execute command for applications) for performing an application program. It is usually recognized in the portion of a command name whether it is a command of which kind. When the given command is an execute command for OS program, the arithmetic processing means 120, This is performed based on OS program in the 1st field M1, and when the given command is an execute command for applications, this is performed based on the application program in the 2nd field M2. Although two or more application program A-E is stored in the 2nd field M2, One of these is specified as current application using the current application specification information stored in the 5th field M5, and the given execute command for applications will be executed by current application.

[0020]Thus, in the case of the conventional IC card, the execute command for applications, It is a command on condition of performing with the application program specified as current application at present, and has become a command on condition of a single application program. For example, when the application program A is specified as current application, from the external device 200, this application program A needs to give the execute command for applications in which interpretation execution is possible. If another word is carried out, in the state of specifying the application program A as current application, it can perform except the execute command for applications which the application program A can interpretation execute.

[0021]Even if the fundamental focus of this invention is in the state where the application program A is specified as current application, For example, if processing using the routine in the application program B can be performed, it is substantially based on technical idea of the ability to make the function of the application program A extend. thus, if ** can carry out between two or more applications for obtaining and a command can be processed, it will become possible to process the work which cannot respond in single application by the coordinated movements of two or more applications.

[0022]For example, the juniper which was a program which performs processing for the application program A to use this IC card as an examination card for hospitals. If the case where this IC card is used as an examination card for hospitals is considered here using the external device 200 installed in the hospital, Naturally, the application program A will be specified as current application, and the various processings for medical examination will be performed based on the program A. And although medical-examination accounting needed to be held by chance, it is assumed that the application program A itself was not equipped with the function which totals the health care cost for one month. In this case, at the conventional IC card, total processing of the health care cost for such one month cannot be performed inside an IC card. However, about the processing which carries out additional installation of the application program B which carries such a summarize function in the IC card concerning this invention, and

totals the health care cost for one month. If a continuous execution rule which uses the routine in this application program B is prepared, total processing within an IC card will be attained. If such a method is taken, handling which extended the function of the application program A will become possible.

[0023]In order to realize such processing, two kinds of commands are prepared for the command which an application program should be made to execute in this invention. The 1st command is a command which should be performed as usual with the single application specified as current application, and is made to call it "the execute command for single applications" here. On the other hand, the 2nd command is a command peculiar to this invention, and it is a command which should be performed using the routine of application programs other than the application specified as current application. Here, it will be called "the continuous execution command for two or more applications." In the case of an above-mentioned example, the processing which totals the health care cost for one month, The application program A specified as current application. It can realize now by giving "the continuous execution command for two or more applications" which performs the application program B different from this continuously (without generating a response in the middle of processing).

[0024]After all, to the IC card concerning this invention, the command of three types as shown in drawing 2 can be given. That is, it is a command of three types, "the execute command for OS program", "the execute command for single applications", and "the continuous execution command for two or more applications."

[0025]Here "the execute command for OS program", As it is a command executed by OS program, for example, is shown in drawing 2. A command like an "application selection command", an "application addition command", and a "continuous execution rule addition command" is prepared (various commands are prepared actually). An "application selection command" is a command for specifying current application, for example, like "SELECT-APPLICATION PROG-A". It becomes a command with the information (PROG-A) which specifies specific application as a command name (SELECT-APPLICATION). If such an "application selection command" is given, the arithmetic processing means 120, Based on OS program, processing which writes the "current application specification information" which shows the specified application program as current application in the 5th field M5 is performed. When the command "SELECT-APPLICATION PROG-A" Becoming is given like the example mentioned above, in the 5th field M5. The specification information on the purport that "the application program A" is current application (for example, "PROG-A" data) will be written in.

[0026]Writing processing of the "current application specification information" on the 5th

field M5 is performed not only when an "application selection command" is given, but when specific current application is specified by directions of OS program. The specific application program by which default specification was carried out specifically at the time of starting of OS program, The "current application specification information" which is specified automatically and specifies the default application program concerned as current application is written in the 5th field M5.

[0027]On the other hand, an "application addition command" and a "continuous execution rule addition command" are "execute commands for OS program" which should be executed by OS program. An "application addition command" is a command for carrying out the additional writing of the application program new to the 2nd field M2, will follow a command name, and the object code of the program concerned will be given from the external device 200 side. The arithmetic processing means 120 performs processing which writes the object code of the given new application in the 2nd field M2 (field which usually consists of EEPROMs) based on OS program. A "continuous execution rule addition command" is a command for carrying out the additional writing of the continuous execution rule new to the 6th field M6, a command name will be followed, and the code which shows the continuous execution rule concerned will be given from the external device 200 side. The arithmetic processing means 120 performs processing which writes the code which shows the given new rule in the 6th field M6 (field which usually consists of EEPROMs) based on OS program.

[0028]In the above, an example of the "execute command for OS program" used by the IC card concerning this embodiment was explained. Usually, in addition, although various "execute commands for OS program" is prepared, detailed explanation is omitted here.

[0029]On the other hand, "the execute command for single applications" is a command which the specific application program specified as current application should be made to execute, and is equivalent to the general execute command for applications in the conventional IC card. A predetermined command will be prepared for every application program like the example of a graphic display with the command for program A, the command for program B, the command for program C, and --. Although it becomes settled depending on each application program, what kind of command is prepared concretely, For example, the command which writes a specific user datum in the 3rd field M3, The command which reads a specific user datum from the 3rd field M3, the command which searches a specific user datum, the command which performs a predetermined operation between the given data and the read user datum, and returns the result as a response, etc. are of infinite variety. Of course, it is also possible to prepare the program A and the command common to the object for B.

[0030]Anyway, this "execute command for single applications", With the single

application program specified as current application, when a command is given, are a command by which executive operation is carried out, and after execution, A certain response is returned to the external device 200 (if a certain data may be transmitted to the external device 200, the check which only shows that processing carried out normal termination may be transmitted). The application program specified as current application, When the given command cannot be executed, the response which shows an error will be returned to the external device 200 (for example, when the command for program B is given in the state where the program A is specified as current application). [0031]"The continuous execution command for two or more applications" used as the feature of this invention, Only with the single application program specified as current application when a command is given. It is a command which cannot carry out executive operation, and will be called the command which can be processed by executing current application and the other application program continuously. For example, after performing processing by the program A specified as current application, a command which processing completes turns into "a continuous execution command for two or more applications" used as the feature of this invention by executing another application program B. If such a command is used and it will see from the external device 200 side, handling by which a new function (function of the program B) was added to the application program A will become possible.

[0032]But in order to make the arithmetic processing means 120 execute such "a continuous execution command for two or more applications", it is necessary to prepare a continuous execution rule for the 6th field M6 beforehand. If another word is carried out, the arithmetic processing means 120 will perform processing which performs two or more routines which straddle two or more applications in the given order based on the continuous execution rule stored in this 6th field M6, when "the continuous execution command for two or more applications" is given. Two or more continuous execution rules are stored in the 6th field M6, and it is made to include practically the information which specifies whether it is a command which uses which continuous execution rule in "the continuous execution command for two or more applications." Although the state where the three rules R1, R2, and R3 are stored in the 6th field M6 is shown in drawing 1, In this case, as "a continuous execution command for two or more applications", as shown in the lower column of drawing 2, three kinds of commands, the command based on the rule R1, the command based on the rule R2, and the command based on the rule R3, can be used. If it is made to include the information which shows any one of the rules R1-R3 to the portion of a command name, and the portion of a parameter in each command, the arithmetic processing means 120 can recognize according to any of the rules R1-R3 it should process, when the command concerned is received.

[0033]Drawing 3 is a block diagram showing an example of the rules R1-R3. The arithmetic processing means 120 will perform processing of "the continuous execution command for two or more applications" in the given order based on such a rule.

[0034]For example, the rule R1 is a rule which shows what the program B should be executed for, after executing the program A. Usually, the programs A and B comprise a set of the routine for executing the command accompanied by a certain data and parameter. Therefore, the rule R1 will include actually the "command for program A" which should be given to the program A, and the "command for program B" which should be given to the program B. In the conventional IC card, when "the command for program A" is given to the arithmetic processing means 120 as an execute command for single applications, after this command is executed by the program A, that result will be returned to the external device 200 side as a response. On the other hand, when "the continuous execution command for two or more applications" based on the rule 1 as shown in drawing 3 is given in the IC card concerning this invention, After the predetermined "command for program A" provided in this rule 1 was handed over by the program A and executed, Then, predetermined "command for program B" provided in this rule 1 will be handed over by the program B, and will be executed, and that result will be returned to the external device 200 side as a response. That is, predetermined processing by the program A and predetermined processing by the program B will be carried out continuously, and the result of this continuous processing will be returned as a response. And the parameter X generated by execution of the program A is handed over to the program B at this time. After all, using the result of processing of the program A, the program B will continue and will be executed.

[0035]It is a program which performs processing for the application program A to use an IC card as an examination card for hospitals like the example mentioned above, Like [the application program B is a program with a total processing capability, and / when performing total processing of the health care cost for one month using the manipulation routine in the application program B] the rule R1, Processing which hands over the parameter X from the program A to the program B can be used. In this case, what is necessary is to define the command "total command of the health care cost for one month" Becoming as "a continuous execution command for two or more applications", and just to define the rule R1 for executing this command as a continuous execution rule. Specifically perform first the read-out command which reads the health care cost list of [for one month] to the program A, and take out the read health care cost list as the parameter X, then the program B is received, What is necessary is to perform the total command which gives this parameter X and performs that total, and just to define the rule R1 of making that result into a response. Though the function to perform total

processing of the health care cost for one month from the first was not prepared for the program A, since only the portion of total processing can be carried out using the routine of the program B, If it sees from the external device 200 side, handling by which a new summarize function was added to the program A will become possible.

[0036]Although the rule R1 shown in drawing 3 is a rule which performs the two application programs A and B in order, it is also possible to, define a rule to which sequential execution of the three or more application programs is carried out, of course. It is not necessary to necessarily define the execution sequence in a program unit, and the continuous execution rule by which the execution sequence about the specific routine in one certain application program was provided may be defined. The rule R2 of drawing 3 is the example which defined the execution rule in the routine unit in this way. That is, the specific routine c1 in the program C is performed first (if needed.). That a certain parameter is given and it may be made to perform the routine c1, Using the parameter X obtained as a result, execute the program D and it continues, The specific routine e2 within the program E will be performed using the parameter Y obtained as a result, the specific routine c2 within the program C will be further performed again using the parameter Z obtained as a result, and processing which returns the result as a response will be performed.

[0037]Generally, since one application program is an aggregate of two or more routines, it can be called rule the continuous execution rule in this invention indicates after all what two or more routines which straddled two or more application programs should be performed for in the given order to be. Therefore, in an application concerned, it will use also in the meaning which it not only uses in the meaning which shows each routine which constitutes a program for the wording a "routine", but shows the whole program which is an aggregate of a routine.

[0038]A continuous execution rule may be a rule which changes the routine which should be followed and performed based on the parameter which does not have to make it the rule which defined the routine which should not necessarily always be performed in a fixed order, and is generated by execution of one routine. In this case, what is necessary is to set two or more candidates as a routine which should be followed and performed, and just to prepare the rule which provided which should be chosen and performed out of two or more of these candidates according to the value of a parameter. The rule R3 of drawing 3 is an example of such a rule. In this example, as a routine which follows the program E, the program B. Three kinds of candidates called the routine b8 within the program D and the program B are set, and based on the parameter X obtained by execution of the program E, any one candidate will follow and will be performed. When the parameter X is positive in the case of the example of a graphic display, the program B

is specifically executed, when the parameter X is negative, the program D is executed, and when the parameter X is 0, the routine b8 within the program B will be performed.

[0039]Although the continuous execution rule prepared in this invention was shown in drawing 3 as three examples, it is possible to define arbitrary rules freely as a continuous execution rule, of course. And as one of "the execute commands for OS program", as shown in the upper column of drawing 2, if the "continuous execution rule addition command" is prepared, as mentioned above, a new continuous execution rule can be added to the 6th field M6, and can be stored if needed.

[0040]Substantial processing by the application program which serves as current application depending on the case may not be performed at all, but only processing by a completely different application program may be performed. For example, when the program A is specified as current application, When "the continuous execution command for two or more applications" to which processing is carried out by only the routine within the program B is given, the program A will not participate in substantial processing at all, but substantial processing will be performed by the program B. However, if it sees from the external device 200 side also in this case and a certain command will be seemingly given to the application program A chosen as current application, it will be dealt with as the response to this has returned. So to speak, the application program A will achieve the function as a window of processing.

[0041]A low thing will be expected if the frequency where "the continuous execution command for two or more applications" is used from a practical viewpoint is measured against the frequency where "the execute command for single applications" is used. In the arithmetic processing means 120 if it stands on such a viewpoint, whenever a certain command will be given from the external device 200, Processing which recognizes each time whether the given command is "an execute command for single applications" and whether it is "a continuous execution command for two or more applications" is performed [inefficient]. Then, practically, if a command is received, the command concerned will be "an execute command for single applications", but the arithmetic processing means 120. Although it is "a continuous execution command for two or more applications", Based on the application program specified at the time as current application, for the time being, without distinguishing both, When processing which tries execution of the given command is performed and an error arises, to recognize the given command to be "a continuous execution command for two or more applications", and what is necessary is just made to perform processing based on the continuous execution rule stored in the 6th field M6.

[0042]As mentioned above, although the portable information processor concerning this invention was explained about the example of the IC card, this invention can apply it

similarly in the portable information processor of not only an IC card but others.

[0043]

[Effect of the Invention]According to this invention, in the portable information processor which stored two or more application programs, a new function can be easily added now about specific application as above.

[Brief Description of the Drawings]

[Drawing 1]It is a block diagram showing the composition of the portable information processor concerning the fundamental embodiment of this invention.

[Drawing 2]It is a figure showing the example of a command three available type in the portable information processor shown in drawing 1.

[Drawing 3]It is a figure showing the example of the continuous execution rule stored in the portable information processor shown in drawing 1.

[Description of Notations]

100 -- Portable information processor (IC card)

110 -- Memory measure

120 -- Arithmetic processing means

130 -- Transmission and reception means

200 -- External device (reader writer device)

A-E -- Application program

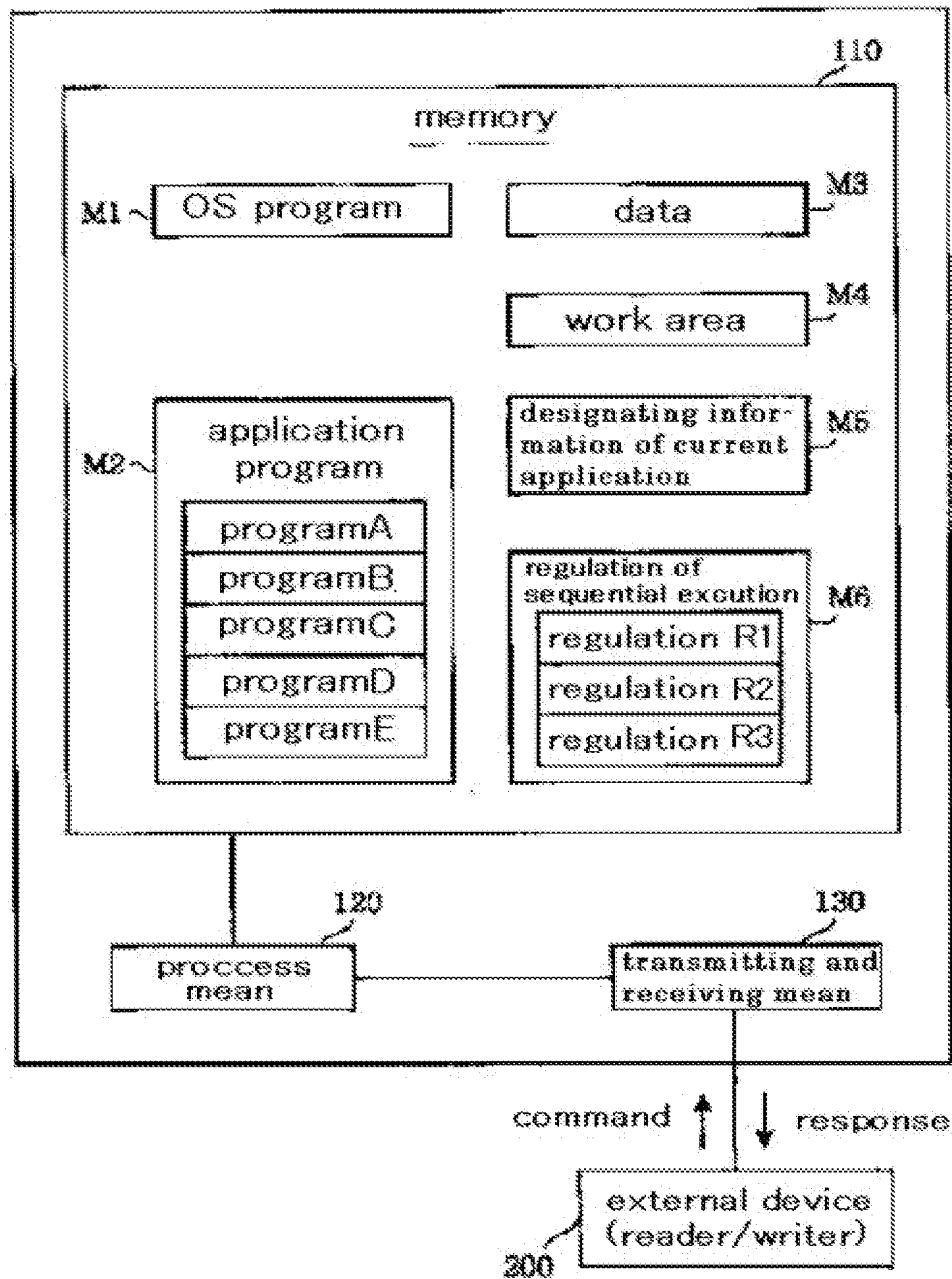
M1-M6 -- Field within the memory measure 110

R1-R3 -- Continuous execution rule

X, Y, Z -- Parameter

【Fig.1】

100 : IC card

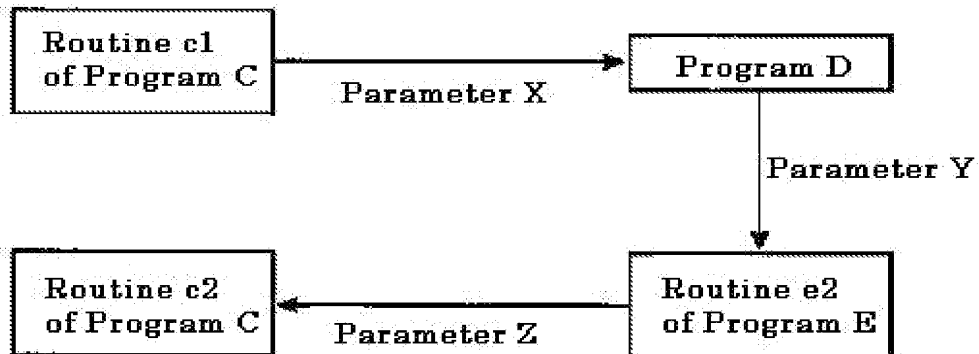


【 Fig. 3 】

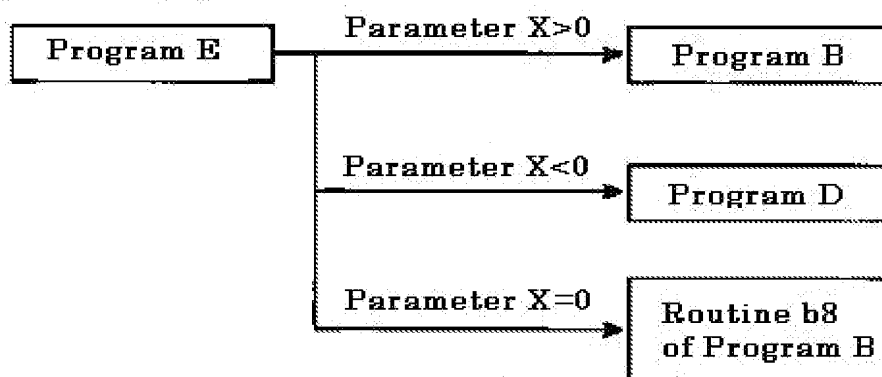
Regulation R1



Regulation R2



Regulation R3



(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号
特開2003-30609
(P2003-30609A)

(43) 公開日 平成15年1月31日 (2003.1.31)

(51) Int.Cl.⁷

識別記号

F I

データベース* (参考)

G 0 6 K 19/07

G 0 6 K 19/00

N 5 B 0 3 j

G 0 6 F 9/445

G 0 6 F 9/06

6 j 0 C 5 B 0 7 6

審査請求 未請求 請求項の数 6 O L (全 10 頁)

(21) 出願番号 特願2001-212023(P2001-212023)

(22) 出願日 平成13年7月12日 (2001.7.12)

(71) 出願人 000002897

大日本印刷株式会社

東京都新宿区市谷加賀町一丁目1番1号

(72) 発明者 神力 哲夫

東京都新宿区市谷加賀町一丁目1番1号

大日本印刷株式会社内

(72) 発明者 入澤 和義

東京都新宿区市谷加賀町一丁目1番1号

大日本印刷株式会社内

(74) 代理人 100091476

弁理士 志村 浩

最終頁に続く

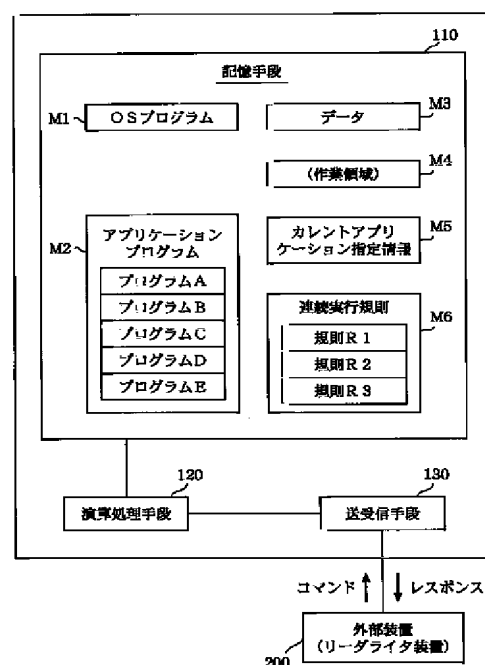
(54) 【発明の名称】 携帯型情報処理装置

(57) 【要約】

【課題】 ICカードに格納された特定のアプリケーションについて、新機能を容易に追加する。

【解決手段】 ICカード100内の領域M2に、複数のアプリケーションA～Eを用意し、外部装置200側からアプリケーション選択コマンドを与えることにより、カレントアプリケーションを指定し、指定情報を領域M5に記録する。通常の実行コマンドは、OSプログラムからカレントアプリケーションに引き渡され実行される。特別なコマンドとして、複数のアプリケーションに跨がって実行される連続実行コマンドを用意し、各連続実行コマンドごとに、領域M6内に連続実行規則を格納しておく。連続実行コマンドが与えられた場合、連続実行規則に基づいて、複数のアプリケーションに跨がったルーチンが所定の順序で実行される。

100: 携帯型情報処理装置 (ICカード)



【特許請求の範囲】

【請求項1】 携帯時には単体として持ち運びされ、アクセス時には外部装置から与えた所定のコマンドに対して所定のレスポンスが返されるように構成された携帯型情報処理装置において、

OSプログラムが記録された第1の領域と、複数のアプリケーションプログラムが記録された第2の領域と、データを格納する第3の領域と、作業領域として用いられる第4の領域と、前記複数のアプリケーションプログラムのの中から選択されたカレントアプリケーションを指定するカレントアプリケーション指定情報を格納する第5の領域と、前記複数のアプリケーションプログラムに跨がった複数のルーチンを所定の順序で実行すべきことを示す連続実行規則を格納する第6の領域と、を有する記憶手段と、

前記OSプログラムの管理下で前記アプリケーションプログラムを実行する演算処理手段と、

前記外部装置から与えられたコマンドを受信して前記演算処理手段に引き渡すとともに、前記演算処理手段から与えられたレスポンスを前記外部装置へ送信する処理を行う送受信手段と、

を備え、前記演算処理手段は、

カレントアプリケーションを指定するためのアプリケーション選択コマンドが与えられたとき、または、OSプログラムの指示により特定のカレントアプリケーションが指定されたときに、該カレントアプリケーションを示すカレントアプリケーション指定情報を前記第5の領域に格納する処理を行い、

単一アプリケーション用実行コマンドが与えられたときに、前記第5の領域に格納されているカレントアプリケーション指定情報によって、その時点においてカレントアプリケーションとして指定されているアプリケーションプログラムに基づいて、与えられたコマンドを実行する処理を行い、

複数アプリケーション用連続実行コマンドが与えられたときに、前記第6の領域に格納されている連続実行規則に基づいて、複数のルーチンを所定の順序で実行する処理を行う機能を有することを特徴とする携帯型情報処理装置。

【請求項2】 請求項1に記載の携帯型情報処理装置において、

連続実行規則が、第1のルーチンの実行後に第2のルーチンを実行すべき旨の順序を示す規則に加えて、第1のルーチンの実行により生成されるパラメータを、第2のルーチンに引き渡す旨の規則を含んでいることを特徴とする携帯型情報処理装置。

【請求項3】 請求項2に記載の携帯型情報処理装置において、

連続実行規則が、第2のルーチンとなるべき複数の候補を定め、第1のルーチンの実行により生成されるパラメ

ータに基づいて、前記複数の候補の中の1つのルーチンを第2のルーチンとして選択しこれを実行すべき旨の規則を含んでいることを特徴とする携帯型情報処理装置。

【請求項4】 請求項1～3のいずれかに記載の携帯型情報処理装置において、

演算処理手段が、送受信手段から単一アプリケーション用実行コマンドまたは複数アプリケーション用連続実行コマンドが与えられたときに、両者を区別せずに、その時点においてカレントアプリケーションとして指定されているアプリケーションプログラムに基づいて、与えられたコマンドの実行を試みる処理を行い、エラーが生じた場合には、与えられたコマンドを複数アプリケーション用連続実行コマンドと認識して、連続実行規則に基づく処理を実行することを特徴とする携帯型情報処理装置。

【請求項5】 請求項1～4のいずれかに記載の携帯型情報処理装置において、

第6の領域に複数の連続実行規則が格納されており、演算処理手段は、与えられた複数アプリケーション用連続実行コマンドに対応した連続実行規則に基づいて、複数のルーチンを所定の順序で実行する処理を行うことを特徴とする携帯型情報処理装置。

【請求項6】 請求項5に記載の携帯型情報処理装置において、

演算処理手段が、新たな連続実行規則を含む連続実行規則追加コマンドが与えられたときに、第6の領域に前記新たな連続実行規則を格納する処理を行うことを特徴とする携帯型情報処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、ICカードのような携帯型情報処理装置に関し、特に、携帯時には単体として持ち運びされ、アクセス時には外部装置から与えた所定のコマンドに対して所定のレスポンスが返されるように構成された携帯型情報処理装置に関する。

【0002】

【従来の技術】電子機器の小型化が進むに従って、一般社会における携帯型情報処理装置の担う役割も益々重要になってきている。特に、小型の携帯型情報処理装置の代表とも言うべきICカードは、近年、様々な分野において実用化されてきており、今後も広く普及するものと予想される。このICカードのような携帯型情報処理装置は、通常は、身体あるいは車両などに伴って携帯され、必要に応じて、パソコンなどに接続されたリーダライタ装置と交信させることによって、外部からのアクセスが可能になる。

【0003】現在、一般に普及しているICカードは、CPUを内蔵したタイプのものであり、内部メモリへのアクセスは、内蔵されたCPUを介してのみ可能となっている。このため、実用上十分なセキュリティが確保さ

れている。一般に、ＩＣカードと外部装置との間のやりとりは、外部装置側からＩＣカードに対してコマンドを送信し、このコマンドに対する返事として、ＩＣカードから外部装置に対してレスポンスを送信する、というプロセスを繰り返すことによって実行される。ＩＣカード内には、通常、情報処理装置としての基本的な機能を統括制御するためのＯＳプログラムと、このＯＳプログラムの管理下で実行されるアプリケーションプログラムとが用意される。ＯＳプログラムもアプリケーションプログラムも、外部装置から与えられたコマンドに基づいて何らかの処理を実行し、何らかのレスポンスを返す処理を行うプログラムであるという点では共通する。

【０００４】最近、ＩＣカードに内蔵されるメモリの記憶容量も増加してきており、ＯＳプログラムの他に、複数のアプリケーションプログラムが格納されたＩＣカードも広く利用されるようになってきている。また、このようなアプリケーションプログラムをＥＥＰＲＯＭなどの書き込み可能な不揮発性メモリに格納できるようにしておき、必要なアプリケーションプログラムを後から追加インストールするような利用形態も普及してきている。このように、複数のアプリケーションプログラムが用意されたＩＣカードに対して、コマンドを与えて何らかの処理を実行させる場合、予め、当該ＩＣカードに対してアプリケーション選択コマンドを与えて特定のアプリケーションを準備状態にする処理が実行される。一般に、アプリケーション選択コマンドによって準備状態となったアプリケーションは、「カレントアプリケーション」と呼ばれている。外部装置側から与えられるアプリケーション用の実行コマンドは、ＯＳプログラムによって、その時点で「カレントアプリケーション」として準備状態になっているアプリケーションプログラムに引き渡されて実行される。

【０００５】

【発明が解決しようとする課題】上述したように、ＩＣカード内に複数のアプリケーションプログラムをインストールしておくことにより、１枚のＩＣカードを種々の用途に利用することができる。また、必要に応じて、新たなアプリケーションプログラムを追加インストールすることも可能であるため、新たな用途を追加することもできる。しかしながら、従来のＩＣカードでは、既にインストールされているアプリケーションプログラムに、付加的な機能を追加することは困難である。これは、１つのアプリケーションプログラムをインストールすると、当該アプリケーションのためのプログラム領域、データ領域などがメモリ上に確定するため、新たな機能のためのルーチンを付加することが困難になるためである。したがって、従来のＩＣカードでは、１つのアプリケーションに付加機能を追加する必要が生じた場合、アプリケーションプログラムごと新規なプログラムに入れ替えざるを得ず、大掛かりな改訂作業が必要になる。

【０００６】そこで本発明は、複数のアプリケーションプログラムを格納した携帯型情報処理装置において、特定のアプリケーションについて、新機能を容易に追加することができる新規な構成を提供することを目的とする。

【０００７】

【課題を解決するための手段】(1) 本発明の第１の態様は、携帯時には単体として持ち運びされ、アクセス時には外部装置から与えた所定のコマンドに対して所定のレスポンスが返されるように構成された携帯型情報処理装置において、ＯＳプログラムが記録された第１の領域と、複数のアプリケーションプログラムが記録された第２の領域と、データを格納する第３の領域と、作業領域として用いられる第４の領域と、複数のアプリケーションプログラムの中から選択されたカレントアプリケーションを指定するカレントアプリケーション指定情報を格納する第５の領域と、複数のアプリケーションプログラムに跨った複数のルーチンを所定の順序で実行すべきことを示す連続実行規則を格納する第６の領域と、を有する記憶手段と、ＯＳプログラムの管理下でアプリケーションプログラムを実行する演算処理手段と、外部装置から与えられたコマンドを受信して演算処理手段に引き渡すとともに、演算処理手段から与えられたレスポンスを外部装置へ送信する処理を行う送受信手段と、を設け、演算処理手段は、カレントアプリケーションを指定するためのアプリケーション選択コマンドが与えられたとき、または、ＯＳプログラムの指示により特定のカレントアプリケーションが指定されたときに、該カレントアプリケーションを示すカレントアプリケーション指定情報を第５の領域に格納する処理を行い、単一アプリケーション用実行コマンドが与えられたときに、第５の領域に格納されているカレントアプリケーション指定情報によって、その時点においてカレントアプリケーションとして指定されているアプリケーションプログラムに基づいて、与えられたコマンドを実行する処理を行い、複数アプリケーション用連続実行コマンドが与えられたときに、第６の領域に格納されている連続実行規則に基づいて、複数のルーチンを所定の順序で実行する処理を行う機能を有するようにし、別なアプリケーションプログラムのルーチンを利用することにより、付加的な機能を実現できるようにしたものである。

【０００８】(2) 本発明の第２の態様は、上述の第１の態様に係る携帯型情報処理装置において、連続実行規則として、第１のルーチンの実行後に第２のルーチンを実行すべき旨の順序を示す規則に加えて、第１のルーチンの実行により生成されるパラメータを、第２のルーチンに引き渡す旨の規則を用いるようにしたものである。

【０００９】(3) 本発明の第３の態様は、上述の第２の態様に係る携帯型情報処理装置において、連続実行規則として、第２のルーチンとなるべき複数の候補を定

め、第1のルーチンの実行により生成されるパラメータに基づいて、複数の候補の中の1つのルーチンを第2のルーチンとして選択しこれを実行すべき旨の規則を用いるようにしたものである。

【0010】(4) 本発明の第4の態様は、上述の第1～第3の態様に係る携帯型情報処理装置において、演算処理手段が、送受信手段から単一アプリケーション用実行コマンドまたは複数アプリケーション用連続実行コマンドが与えられたときに、両者を区別せずに、その時点においてカレントアプリケーションとして指定されているアプリケーションプログラムに基づいて、与えられたコマンドの実行を試みる処理を行い、エラーが生じた場合には、与えられたコマンドを複数アプリケーション用連続実行コマンドと認識して、連続実行規則に基づく処理を実行するようにしたものである。

【0011】(5) 本発明の第5の態様は、上述の第1～第4の態様に係る携帯型情報処理装置において、第6の領域に複数の連続実行規則を格納するようにし、演算処理手段が、与えられた複数アプリケーション用連続実行コマンドに対応した連続実行規則に基づいて、複数のルーチンを所定の順序で実行する処理を行うことができるようにしたものである。

【0012】(6) 本発明の第6の態様は、上述の第5の態様に係る携帯型情報処理装置において、演算処理手段が、新たな連続実行規則を含む連続実行規則追加コマンドが与えられたときに、第6の領域に新たな連続実行規則を格納する処理を行うことができるようにしたものである。

【0013】

【発明の実施の形態】以下、本発明を図示する実施形態に基づいて説明する。図1は、本発明の一実施形態に係る携帯型情報処理装置（具体的には、ICカード）100の構成を示すブロック図である。この携帯型情報処理装置100は、携帯時には単体として持ち運びされ、アクセス時には外部装置（具体的には、リーダライタ装置）200に接続される。この外部装置200から所定のコマンドを与えると、携帯型情報処理装置100内部において所定の処理が実施され、その結果がレスポンスとして外部装置200へと返される。

【0014】携帯型情報処理装置100の基本構成要素は、図示のとおり、記憶手段110、演算処理手段120、送受信手段130である。記憶手段110には、種々のプログラムやデータが記録される。演算処理手段120は、この記憶手段110内のプログラムに基づいて、外部装置200から与えられるコマンドを実行する。外部装置200に対する送受信処理は、送受信手段130によって行われる。送受信手段130と外部装置200との間の通信は、物理的に電極を接触させて電気信号をやりとりする接触方式で行ってもよいし、電磁的な信号路を形成して情報をやりとりする非接触方式で行

ってもよい。外部装置200から与えられたコマンドは、送受信手段130で受信され、演算処理手段120によって実行される。コマンドの実行後、演算処理手段120から何らかのレスポンスが出力され、送受信手段130によって外部装置200へと送信される。このように、携帯型情報処理装置100と外部装置200との間の通信は、外部装置200からのコマンド送信と携帯型情報処理装置100からのレスポンス送信を交互に繰り返すことにより行われる。

【0015】記憶手段110内には、図示のとおり、6つの領域が設けられている。第1の領域M1には、この携帯型情報処理装置100による処理動作を統括管理するためのOSプログラムが格納されている。演算処理手段120は、基本的には、このOSプログラムの管理下で種々の処理を実行することになる。第2の領域M2には、複数のアプリケーションプログラムが格納されている。図では、具体的に、5種類のアプリケーションプログラムA～Eが用意されている例が示されている。通常、これらの各アプリケーションプログラムは、それぞれ異なる用途に利用されるプログラムになる。たとえば、プログラムAは、このICカードを病院用診察カードとして利用するための処理を行うプログラムであり、プログラムBは、このICカードをクレジットカードとして利用するための処理を行うプログラムであり、プログラムCは、このICカードを図書館用貸出カードとして利用するための処理を行うプログラムであり、…といった具合である。

【0016】第3の領域M3には、データが格納されている。このデータは、OSプログラムや各アプリケーションプログラムが利用するためのデータであり、複数のプログラムから共通して利用されるデータもあれば、個々のアプリケーションプログラムに固有のデータもある。なお、実用上は、特定のアプリケーションプログラムに固有のデータは、当該アプリケーションプログラムの格納領域に隣接した領域に格納するのが一般的であるが、ここでは便宜上、いずれのアプリケーションで用いられるデータも、第3の領域M3内に格納されるものとして以下の説明を行う。また、第4の領域M4は、演算処理手段120が種々の演算処理を行う上での作業領域として利用される領域であり、第5の領域M5は、カレントアプリケーション指定情報を格納するための領域である。ここで、「カレントアプリケーション」とは、第2の領域M2に格納されている複数のアプリケーションプログラムA～Eの中から選択された1つのアプリケーションであり、「現時点で選択されているアプリケーション」ということになる。外部装置200から与えられたコマンドは、原則的に（後述するように、単一アプリケーション用実行コマンドである限りは）、この「カレントアプリケーション」によって実行される。なお、この第5の領域M5は、第4の領域M4の一部として取り

扱うことも可能であるが、本発明においては重要な構成要素であるため、ここでは「カレントアプリケーション指定情報」を格納するための専用の領域として、第5の領域M5を定義することにする。

【0017】上述した第1の領域M1～第5の領域M5は、従来のICカードにおいても利用されている領域である。本発明の特徴は、更に、第6の領域M6を設け、ここに連続実行規則R1～R3を格納するようにし、第2の領域M2内に用意されている複数のアプリケーションプログラムに跨がった複数のルーチンを実行できるようにした点にある。ここで、連続実行規則R1～R3は、後に詳述するように、それぞれ複数のアプリケーションプログラムに跨がった複数のルーチンを所定の順序で実行すべきことを示す規則であり、演算処理手段120は、この連続実行規則に基づいて、外部から与えられたコマンドを複数のアプリケーションに処理させることができる。

【0018】なお、図1に示す実施形態では、記憶手段110を1つのブロックで示しているが、実際には、ICカード内にはROM、EEPROMなどの不揮発性メモリと、RAMなどの揮発性メモリとが内蔵されており、各領域M1～M6は、これら複数のメモリに適宜分散して設けられている。この実施形態では、OSプログラムを格納する第1の領域M1はROM内に設けられている（一部は、EEPROM内に設けて書き替えができるようにしてもよい）。また、作業領域として利用される第4の領域M4およびカレントアプリケーション指定情報を格納する第5の領域M5は、RAM内に設けられている。このICカードでは、電力はすべて外部装置200側から供給されるので、外部装置200との通信が完了し、外部装置200が切り離されると、第4の領域M4および第5の領域M5の内容はすべて失われることになる。一方、アプリケーションプログラムを格納する第2の領域M2、データを格納する第3の領域M3、連続実行規則を格納する第6の領域M6は、いずれもEEPROM内に設けられている。したがって、これらの情報は、外部装置200が切り離された後もそのまま保持されることになる。また、EEPROMは書換えや追加書込が可能であるため、アプリケーションプログラム、データ、連続実行規則は、いずれも修正したり追加したりすることができる。

【0019】演算処理手段120は、ICカード用のCPUから構成されており、OSプログラムの管理下で、アプリケーションプログラムA～Eを実行する機能を有している。上述したように、外部装置200から与えられたコマンドは、送受信手段130を介して演算処理手段120に引き渡される。一般に、コマンドは、OSプログラムに実行させるためのコマンド（以下、OSプログラム用実行コマンドという）と、アプリケーションプログラムに実行させるためのコマンド（以下、アプリケ

ーション用実行コマンドという）とに分類できる。いずれの種類のコマンドであるかは、通常、コマンド名の部分で認識する。演算処理手段120は、与えられたコマンドがOSプログラム用実行コマンドであった場合には、第1の領域M1内のOSプログラムに基づいてこれを実行し、与えられたコマンドがアプリケーション用実行コマンドであった場合には、第2の領域M2内のアプリケーションプログラムに基づいてこれを実行する。第2の領域M2内には、複数のアプリケーションプログラムA～Eが格納されているが、このうちの1つが、第5の領域M5に格納されているカレントアプリケーション指定情報によって、カレントアプリケーションとして指定されており、与えられたアプリケーション用実行コマンドは、カレントアプリケーションによって実行されることになる。

【0020】このように、従来のICカードの場合、アプリケーション用実行コマンドは、現時点でカレントアプリケーションとして指定されているアプリケーションプログラムによって実行されることを前提としたコマンドであり、単一のアプリケーションプログラムを前提としたコマンドとなっている。たとえば、アプリケーションプログラムAがカレントアプリケーションとして指定されていた場合、外部装置200からは、このアプリケーションプログラムAが解釈実行可能なアプリケーション用実行コマンドを与える必要がある。別言すれば、アプリケーションプログラムAをカレントアプリケーションとして指定している状態では、アプリケーションプログラムAが解釈実行可能なアプリケーション用実行コマンド以外は実行することができないことになる。

【0021】本発明の基本的な着眼点は、アプリケーションプログラムAがカレントアプリケーションとして指定されている状態であっても、たとえば、アプリケーションプログラムB内のルーチンを利用した処理を行うことができれば、実質的に、アプリケーションプログラムAの機能を拡張させることができるであろう、という技術思想に基づくものである。このように、複数のアプリケーション間を跨がうようにしてコマンドを処理することができれば、単一のアプリケーションでは対応できないような仕事を、複数のアプリケーションの連携動作により処理することが可能になる。

【0022】たとえば、アプリケーションプログラムAが、このICカードを病院用診察カードとして利用するための処理を行うプログラムであったとしよう。ここで、病院に設置された外部装置200を用いて、このICカードを病院用診察カードとして利用する場合を考えると、当然、アプリケーションプログラムAがカレントアプリケーションとして指定され、プログラムAに基づいて診療のための種々の処理が実行されることになる。そして、たまたま診療会計処理を行う必要が生じたが、アプリケーションプログラムA自身には、1か月間の医

療費を集計する機能が備わっていなかったとする。この場合、従来のICカードでは、このような1か月間の医療費の集計処理を、ICカード内部で実行させることはできない。ところが本発明に係るICカードでは、このような集計機能を搭載したアプリケーションプログラムBを追加インストールし、1ヶ月間の医療費を集計する処理については、このアプリケーションプログラムB内のルーチンを利用するような連続実行規則を用意すれば、ICカード内での集計処理が可能になる。このような方法をとれば、あたかもアプリケーションプログラムAの機能を拡張したような取り扱いが可能になる。

【0023】このような処理を実現するために、本発明では、アプリケーションプログラムに実行させるべきコマンドに、2通りのコマンドを用意している。第1のコマンドは、従来どおり、カレントアプリケーションとして指定されている単一のアプリケーションによって実行させるべきコマンドであり、ここでは「単一アプリケーション用実行コマンド」と呼ぶことにする。一方、第2のコマンドは、本発明に特有のコマンドであり、カレントアプリケーションとして指定されているアプリケーション以外のアプリケーションプログラムのルーチンを利用して実行させるべきコマンドであり、ここでは「複数アプリケーション用連続実行コマンド」と呼ぶことにする。上述の例の場合、1か月間の医療費を集計する処理は、カレントアプリケーションとして指定されているアプリケーションプログラムAと、これとは別なアプリケーションプログラムBとを連続的に（処理途中でレスポンスを発生させることなく）実行させる「複数アプリケーション用連続実行コマンド」を与えることにより実現できるようになる。

【0024】結局、本発明に係るICカードに対しては、図2に示すような3つのタイプのコマンドを与えることができる。すなわち、「OSプログラム用実行コマンド」、「単一アプリケーション用実行コマンド」、「複数アプリケーション用連続実行コマンド」の3つのタイプのコマンドである。

【0025】ここで、「OSプログラム用実行コマンド」は、OSプログラムによって実行されるコマンドであり、たとえば、図2に示すように、「アプリケーション選択コマンド」、「アプリケーション追加コマンド」、「連続実行規則追加コマンド」のようなコマンドが用意される（実際には、この他にも種々のコマンドが用意される）。「アプリケーション選択コマンド」は、カレントアプリケーションを指定するためのコマンドであり、たとえば「SELECT-APPLICATION PROG-A」のように、コマンド名（SELECT-APPLICATION）と、特定のアプリケーションを指定する情報（PROG-A）をもったコマンドになる。演算処理手段120は、このような「アプリケーション選択コマンド」が与えられると、OSプログラ

ムに基づいて、指定されたアプリケーションプログラムをカレントアプリケーションとして示す「カレントアプリケーション指定情報」を第5の領域M5に書き込む処理を実行する。上述した例のように、「SELECT-APPLICATION PROG-A」なるコマンドが与えられた場合、第5の領域M5には、「アプリケーションプログラムA」がカレントアプリケーションである旨の指定情報（たとえば、「PROG-A」なるデータ）が書き込まれることになる。

【0026】なお、第5の領域M5への「カレントアプリケーション指定情報」の書込処理は、「アプリケーション選択コマンド」が与えられた場合だけでなく、OSプログラムの指示により特定のカレントアプリケーションが指定されたときにも実行される。具体的には、OSプログラムの起動時に、デフォルト指定された特定のアプリケーションプログラムが、カレントアプリケーションとして自動的に指定され、当該デフォルトのアプリケーションプログラムを指定する「カレントアプリケーション指定情報」が第5の領域M5に書き込まれる。

【0027】一方、「アプリケーション追加コマンド」および「連続実行規則追加コマンド」も、OSプログラムによって実行されるべき「OSプログラム用実行コマンド」である。「アプリケーション追加コマンド」は、第2の領域M2に新たなアプリケーションプログラムを追加書込するためのコマンドであり、コマンド名に後続して、当該プログラムのオブジェクトコードが外部装置200側から与えられることになる。演算処理手段120は、OSプログラムに基づいて、与えられた新規アプリケーションのオブジェクトコードを、第2の領域M2（通常は、EEPROMからなる領域）へ書き込む処理を実行する。また、「連続実行規則追加コマンド」は、第6の領域M6に新たな連続実行規則を追加書込するためのコマンドであり、コマンド名に後続して、当該連続実行規則を示すコードが外部装置200側から与えられることになる。演算処理手段120は、OSプログラムに基づいて、与えられた新規規則を示すコードを、第6の領域M6（通常は、EEPROMからなる領域）へ書き込む処理を実行する。

【0028】以上、本実施形態に係るICカードで用いられる「OSプログラム用実行コマンド」の一例を説明した。通常は、この他にも種々の「OSプログラム用実行コマンド」が用意されているが、ここでは詳しい説明は省略する。

【0029】一方、「単一アプリケーション用実行コマンド」は、カレントアプリケーションとして指定されている特定のアプリケーションプログラムに実行させるべきコマンドであり、従来のICカードにおける一般的なアプリケーション用実行コマンドに相当するものである。図示の例のように、プログラムA用コマンド、プログラムB用コマンド、プログラムC用コマンド、…と各

アプリケーションプログラムごとに所定のコマンドが用意されることになる。具体的にどのようなコマンドが用意されるかは、個々のアプリケーションプログラムに依存して定まるが、たとえば、特定のユーザデータを第3の領域M3に書き込むコマンドや、特定のユーザデータを第3の領域M3から読み出すコマンドや、特定のユーザデータを検索するコマンドや、与えられたデータと読み出したユーザデータとの間で所定の演算を実行してその結果をレスポンスとして返すコマンドなど千差万別である。もちろん、プログラムA、B用に共通したコマンドを用意することも可能である。

【0030】いずれにせよ、この「単一アプリケーション用実行コマンド」は、コマンドを与えた時点においてカレントアプリケーションとして指定されている単一のアプリケーションプログラムによって実行処理されるコマンドであり、実行後には、何らかのレスポンスが外部装置200に返される（外部装置200に対して、何らかのデータを送信する場合もあれば、単に、処理が正常終了したことを示す確認を送信する場合もある）。また、カレントアプリケーションとして指定されているアプリケーションプログラムが、与えられたコマンドを実行することができない場合（たとえば、プログラムAがカレントアプリケーションとして指定されている状態において、プログラムB用コマンドが与えられたような場合）には、エラーを示すレスポンスが外部装置200に返されることになる。

【0031】本発明の特徴となる「複数アプリケーション用連続実行コマンド」は、コマンドを与えた時点においてカレントアプリケーションとして指定されている単一のアプリケーションプログラムだけでは、実行処理することができないコマンドであり、カレントアプリケーションとそれ以外のアプリケーションプログラムとを連続的に実行することにより処理することが可能なコマンドということになる。たとえば、カレントアプリケーションとして指定されているプログラムAによる処理を実行した後、更に、別なアプリケーションプログラムBを実行することにより処理が完了するようなコマンドが、本発明の特徴となる「複数アプリケーション用連続実行コマンド」になる。このようなコマンドを用いれば、外部装置200側から見れば、あたかもアプリケーションプログラムAに新たな機能（プログラムBの機能）が追加されたような取り扱いが可能になる。

【0032】もっとも、このような「複数アプリケーション用連続実行コマンド」を演算処理手段120に実行させるためには、予め、第6の領域M6に連続実行規則を用意しておく必要がある。別言すれば、演算処理手段120は、「複数アプリケーション用連続実行コマンド」が与えられた場合、この第6の領域M6に格納されている連続実行規則に基づいて、複数のアプリケーションに跨がる複数のルーチンを所定の順序で実行する処理

を行うことになる。実用上は、第6の領域M6には、複数の連続実行規則を格納しておくようにし、「複数アプリケーション用連続実行コマンド」には、どの連続実行規則を用いるコマンドであるかを指定する情報を含ませておくようにする。図1には、第6の領域M6内に3つの規則R1、R2、R3が格納されている状態が示されているが、この場合は、「複数アプリケーション用連続実行コマンド」としては、図2の下欄に示すように、規則R1に基づくコマンド、規則R2に基づくコマンド、規則R3に基づくコマンドの3通りのコマンドを用いることができる。各コマンドに、コマンド名の部分やパラメータの部分に、規則R1～R3のいずれか1つを示す情報を含ませておくようにすれば、演算処理手段120は、当該コマンドを受け取ったときに、規則R1～R3のいずれに従って処理すべきかを認識することができる。

【0033】図3は、規則R1～R3の一例を示すブロック図である。演算処理手段120は、このような規則に基づいて、「複数アプリケーション用連続実行コマンド」の処理を所定の順序で実行することになる。

【0034】たとえば、規則R1は、プログラムAを実行した後、プログラムBを実行すべきことを示す規則である。通常、プログラムAやBは、何らかのデータやパラメータを伴うコマンドを実行するためのルーチンの集合から構成されている。したがって、実際には、規則R1は、プログラムAに対して与えるべき「プログラムA用コマンド」とプログラムBに対して与えるべき「プログラムB用コマンド」とを含んでいることになる。従来のICカードにおいて、「プログラムA用コマンド」が単一アプリケーション用実行コマンドとして演算処理手段120に与えられた場合、プログラムAによって、このコマンドが実行された後、その結果がレスポンスとして外部装置200側に返されることになる。これに対して、本発明に係るICカードにおいて、図3に示すような規則1に基づく「複数アプリケーション用連続実行コマンド」が与えられた場合は、この規則1に定められている所定の「プログラムA用コマンド」がプログラムAに引き渡されて実行された後、続いて、この規則1に定められている所定の「プログラムB用コマンド」がプログラムBに引き渡されて実行され、その結果が、レスポンスとして外部装置200側に返されることになる。すなわち、プログラムAによる所定の処理とプログラムBによる所定の処理とが連続的に実施され、この連続的な処理の結果がレスポンスとして返されることになる。そして、このとき、プログラムAの実行により生成されるパラメータXが、プログラムBへと引き渡される。結局、プログラムAの処理の結果を利用して、プログラムBが続いて実行されることになる。

【0035】前述した例のように、アプリケーションプログラムAが、ICカードを病院用診察カードとして利

用するための処理を行うプログラムであり、アプリケーションプログラムBが、集計処理機能をもったプログラムであり、1か月間の医療費の集計処理を、アプリケーションプログラムB内の処理ルーチンを利用して行うような場合には、規則R1のように、プログラムAからプログラムBへとパラメータXを引き渡すような処理が利用できる。この場合、「複数アプリケーション用連続実行コマンド」として、「1か月間の医療費の集計コマンド」なるコマンドを定義しておき、このコマンドを実行するための規則R1を連続実行規則として定めておけばよい。具体的には、まず、プログラムAに対して1か月分の医療費リストを読み出す読出コマンドを実行させ、読み出された医療費リストをパラメータXとして取り出し、続いて、プログラムBに対して、このパラメータXを与えてその集計を行う集計コマンドを実行させ、その結果をレスポンスとする、というような規則R1を定めておけばよい。プログラムAにはもともと1か月間の医療費の集計処理を行う機能は用意されていなかったとしても、集計処理の部分だけをプログラムBのルーチンを利用して実施することができるので、外部装置200側から見れば、プログラムAに新たな集計機能が付加されたような取り扱いが可能になる。

【0036】図3に示す規則R1は、2つのアプリケーションプログラムA、Bを順番に実行させる規則であるが、もちろん、3つ以上のアプリケーションプログラムを順次実行させるような規則を定義することも可能である。また、必ずしもプログラム単位での実行順序を定義する必要はなく、ある1つのアプリケーションプログラム中の特定のルーチンについての実行順序を定めた連続実行規則を定義してもかまわない。図3の規則R2は、このようにルーチン単位での実行規則を定めた例である。すなわち、まず、プログラムCの中の特定のルーチンc1を実行し（必要に応じて、何らかのパラメータを与えてルーチンc1を実行するようにしてもよい）、その結果得られるパラメータXを用いてプログラムDを実行し、続いて、その結果得られるパラメータYを用いてプログラムE内の特定のルーチンe2を実行し、更に、その結果得られるパラメータZを用いて再びプログラムC内の特定のルーチンc2を実行し、その結果をレスポンスとして返す処理が行われることになる。

【0037】一般に、1つのアプリケーションプログラムは、複数のルーチンの集合体であるから、結局、本発明における連続実行規則とは、複数のアプリケーションプログラムに跨った複数のルーチンを所定の順序で実行すべきことを示す規則ということができる。したがって、本願では、「ルーチン」という文言を、プログラムを構成する1つ1つのルーチンを示す意味で用いるだけでなく、ルーチンの集合体であるプログラム全体を示す意味でも用いることにする。

【0038】なお、連続実行規則は、必ずしも常に一定

の順序で実行すべきルーチンを定義した規則にする必要はなく、1つのルーチンの実行により生成されるパラメータに基づいて、後続して実行すべきルーチンが変わるような規則であってもかまわない。この場合、後続して実行されるべきルーチンとして複数の候補を定めておき、パラメータの値に応じて、この複数の候補の中からどれを選択して実行すべきかを定めた規則を用意しておけばよい。図3の規則R3は、このような規則の一例である。この例では、プログラムEに後続するルーチンとして、プログラムB、プログラムD、プログラムB内のルーチンb8という3通りの候補が定められており、プログラムEの実行によって得られるパラメータXに基づいて、いずれか1つの候補が後続して実行されることになる。具体的には、図示の例の場合、パラメータXが正の場合は、プログラムBが実行され、パラメータXが負の場合は、プログラムDが実行され、パラメータXが0の場合は、プログラムB内のルーチンb8が実行されることになる。

【0039】図3には、本発明において用意される連続実行規則を、3つの例として示したが、もちろん、連続実行規則としては、任意の規則を自由に定義することが可能である。しかも、図2の上欄に示すように、「OSプログラム用実行コマンド」のひとつとして、「連続実行規則追加コマンド」を用意しておけば、前述したように、必要に応じて、新たな連続実行規則を第6の領域M6に追加して格納することができる。

【0040】また、場合によっては、カレントアプリケーションとなっているアプリケーションプログラムによる実質的な処理は全く行われず、全く別なアプリケーションプログラムによる処理のみが実行される場合もある。たとえば、プログラムAがカレントアプリケーションとして指定されている場合に、プログラムB内のルーチンのみによって処理が行われる「複数アプリケーション用連続実行コマンド」が与えられた場合、プログラムAは実質的な処理には何ら関与せず、実質的な処理はプログラムBのみによって行われることになる。しかし、この場合も、外部装置200側からみれば、見掛け上、カレントアプリケーションとして選択されているアプリケーションプログラムAに対して何らかのコマンドを与えたら、これに対するレスポンスが戻ってきたように取り扱われる。いわば、アプリケーションプログラムAは、処理の窓口としての機能を果たすことになる。

【0041】なお、実用上の観点からは、「複数アプリケーション用連続実行コマンド」を利用する頻度は、「単一アプリケーション用実行コマンド」を利用する頻度に比べれば、低いことが予想される。このような観点に立てば、外部装置200から何らかのコマンドが与えられるたびに、演算処理手段120において、与えられたコマンドが「単一アプリケーション用実行コマンド」であるのか、「複数アプリケーション用連続実行コマン

ド」であるのかをその都度認識する処理を行うのは非効率的である。そこで、実用上は、演算処理手段120は、コマンドを受け取ったら、当該コマンドが、「単一アプリケーション用実行コマンド」であろうが、「複数アプリケーション用連続実行コマンド」であろうが、両者を区別せずに、とりあえず、その時点においてカレントアプリケーションとして指定されているアプリケーションプログラムに基づいて、与えられたコマンドの実行を試みる処理を行い、エラーが生じた場合には、与えられたコマンドを「複数アプリケーション用連続実行コマンド」と認識して、第6の領域M6に格納されている連続実行規則に基いて処理を実行するようにしておけばよい。

【0042】以上、本発明に係る携帯型情報処理装置をICカードの例について説明したが、本発明はICカードに限らず、その他の携帯型情報処理装置においても同様に適用が可能である。

【0043】

【発明の効果】以上のとおり本発明によれば、複数のアプリケーションプログラムを格納した携帯型情報処理装

置において、特定のアプリケーションについて、新機能を容易に追加することができるようになる。

【図面の簡単な説明】

【図1】本発明の基本的な実施形態に係る携帯型情報処理装置の構成を示すブロック図である。

【図2】図1に示す携帯型情報処理装置において利用可能な3つのタイプのコマンドの例を示す図である。

【図3】図1に示す携帯型情報処理装置内に格納される連続実行規則の例を示す図である。

【符号の説明】

100…携帯型情報処理装置（ICカード）

110…記憶手段

120…演算処理手段

130…送受信手段

200…外部装置（リーダライタ装置）

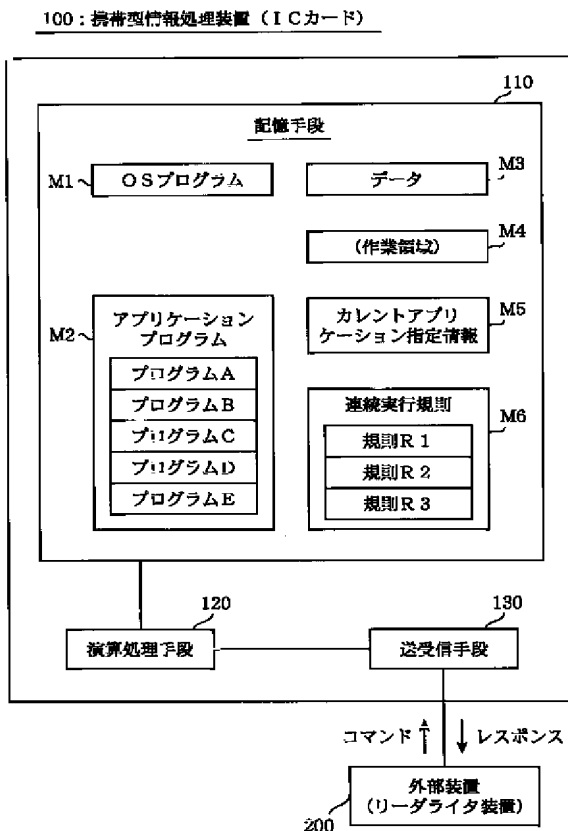
A～E…アプリケーションプログラム

M1～M6…記憶手段110内の領域

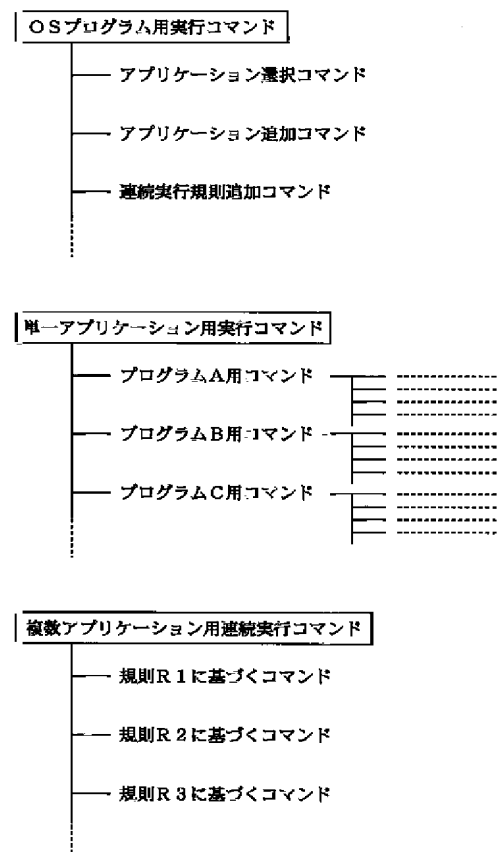
R1～R3…連続実行規則

X, Y, Z…パラメータ

【図1】

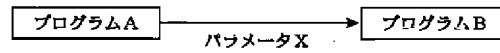


【図2】

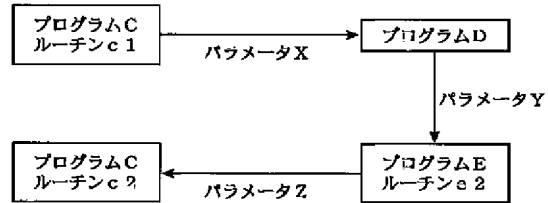


【図3】

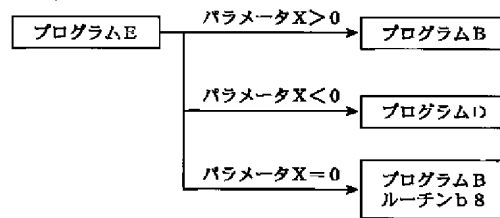
規則R 1



規則R 2



規則R 3



フロントページの続き

(72)発明者 柴田 直人
東京都新宿区市谷加賀町一丁目1番1号
大日本印刷株式会社内

Fターム(参考) 5B035 AA06 BB09 CA11 CA29
5B076 AB17